Assignment – 1

1. **Write a java program to find the maximum & minimum element in an array.**

#include <stdio.h>

#include <stdlib.h>

int main()

{

    int n;

    printf("Enter the number of elements in the array: ");

    scanf("%d", &n);

    int arr[n];

    if (n < 1)

    {

        printf("Invalid input\n");

        exit(0);

    }

    printf("Enter the elements of the array: ");

    for (int i = 0; i < n; i++)

    {

        scanf("%d", &arr[i]);

    }

    int max = arr[0];

    for (int i = 1; i < n; i++)

    {

        if (arr[i] > max)

        {

            max = arr[i];

        }

    }

    printf("The maximum element in the array is: %d\n", max);

    return 0;

}

Source Code



Set 1

Enter the number of elements in the array: 0

Invalid input

Set 2

Enter the number of elements in the array: 5

Enter the elements of the array: 2 1 9 7 3

The maximum element in the array is: 9

Output

1. **Implement a java program to reverse an array.**

#include <stdio.h>

#include <stdlib.h>

int main()

{

    int n;

    printf("Enter the number of elements in the array: ");

    scanf("%d", &n);

    int arr[n];

    if (n < 1)

    {

        printf("Invalid input\n");

        exit(0);

    }

    printf("Enter the elements of the array: ");

    for (int i = 0; i < n; i++)

    {

        scanf("%d", &arr[i]);

    }

    int max = arr[0];

    for (int i = 1; i < n; i++)

    {

        if (arr[i] > max)

        {

            max = arr[i];

        }

    }

    printf("The maximum element in the array is: %d\n", max);

    return 0;

}

Source Code



void reverse(int arr[], int n)

{

    int temp;

    for (int i = 0; i < n / 2; i++)

    {

        temp = arr[i];

        arr[i] = arr[n - i - 1];

        arr[n - i - 1] = temp;

    }

}

Source Code: reverse()

Set 1

Enter the number of elements in the array: 0

Invalid input

Set 2

Enter how many elements you want: 5

Enter the array elements: 1 2 3 4 5

The array is: 1 2 3 4 5

The reverse array is: 5 4 3 2 1

Output

1. **Write a java program to check an array is palindrome or not.**

#include <stdio.h>

#include <stdlib.h>

int main()

{

    int n;

    printf("Enter the number of elements in the array: ");

    scanf("%d", &n);

    int arr[n];

    if (n < 1)

    {

        printf("Invalid input\n");

        exit(0);

    }

    printf("Enter the elements of the array: ");

    for (int i = 0; i < n; i++)

    {

        scanf("%d", &arr[i]);

    }

    int max = arr[0];

    for (int i = 1; i < n; i++)

    {

        if (arr[i] > max)

        {

            max = arr[i];

        }

    }

    printf("The maximum element in the array is: %d\n", max);

    return 0;

}

Source Code



Enter the number of elements in the first array: 5

Enter the elements in the first array: 6 4 5 8 2

Enter the number of elements in the second array: 3

Enter the elements in the second array: 2 4 3

The intersection of the two arrays is: 4 2

Output

1. **Write an algorithm to rotate an array given number of positions.**

#include <stdio.h>

#include <stdlib.h>

int main()

{

    int n;

    printf("Enter the number of elements in the array: ");

    scanf("%d", &n);

    int arr[n];

    if (n < 1)

    {

        printf("Invalid input\n");

        exit(0);

    }

    printf("Enter the elements of the array: ");

    for (int i = 0; i < n; i++)

    {

        scanf("%d", &arr[i]);

    }

    int max = arr[0];

    for (int i = 1; i < n; i++)

    {

        if (arr[i] > max)

        {

            max = arr[i];

        }

    }

    printf("The maximum element in the array is: %d\n", max);

    return 0;

}

Source Code



void rotate(int arr[], int n, int pos)

{

    // Adjust position to be within bounds

    if (pos > n)

        pos = pos % n;

    // Create a temporary array to hold the rotated values

    int temp[max];

    for (int i = 0; i < n; i++)

        temp[(i + pos) % n] = arr[i];

    // Step 3: Copy back from temp to arr

    for (int i = 0; i < n; i++)

        arr[i] = temp[i];

}

Source Code: rotate()

()

Enter how many elements you want: 5

Enter the array elements: 1 2 3 4 5

The position of rotation: 3

The array is: 1 2 3 4 5

The rotated array is: 3 4 5 1 2

Output